Klasifikasi Berita Indonesia Menggunakan Metode Naive Bayesian Classification dan Support Vector Machine dengan Confix Stripping Stemmer

Dio Ariadi dan Kartika Fithriasari
Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember
Jl. Arief Rahman Hakim, Surabaya 60111
e-mail :kartika_f@statistika.its.ac.id

***Abstrak* -- Jumlah aliran artikel berita yang diunggah di internet sangat banyak dan rentang waktu yang cepat. Jumlah yang banyak dan waktu yang cepat akan menyulitkan editor mengkategorikan secara manual. Terdapat metode agar berita dapat dikategorikan secara otomatis, yaitu klasifikasi. Data berita berbentuk teks, sehingga jauh lebih rumit dan perlu proses untuk mempersiapkan data. Salah satu prosesnya adalah confix-stripping stemmer sebagai cara untuk mendapatkan kata dasar dari berita Indonesia. Untuk metode klasifikasi yang digunakan adalah Naive Bayes Classifier (NBC) yang secara umum sering digunakan dalam data teks dan Support Vector Machine (SVM) yang diketahui bekerja sangat baik pada data dengan dimensi besar. Kedua metode tersebut akan dibandingkan untuk mengetahui hasil klasifikasi yang paling baik. Hasil penelitian menunjukkan bahwa SVM kernel Linier dan kernel RBF menghasilkan ketepatan klasifikasi yang sama dan bila dibandingkan dengan NBC maka SVM lebih baik.**

Kata Kunci -- artikel berita, confix-stripping stemmer, klasifikasi, naive bayes classifier, support vector machine

I. PENDAHULUAN

Pada tahun 2006 pertumbuhan dan pertukaran informasi sudah mencapai lebih dari 550 triliun dokumen dan 7,3 juta Internet page baru tiap harinya. Salah satu dampaknya adalah artikel berita yang diunggah di internet sangatlah banyak dan rentang waktu yang cepat. Selama ini pengkategorian berita masih menggunakan tenaga manusia atau manual.Kategori yang banyak beserta waktu yang cepat akan menyulitkan editor untuk mengkategorikan, terutama artikel yang tidak terlalu berbeda secara jelas. Beberapa kategori yang penggunaan bahasanya tidak berbeda terlalu jauh seperti nasional, internasional, sains, ekonomi, tekno, health, dan properti mengharuskan seorang editor mengetahui isi artikel yang akan diunggah secara keseluruhan untuk selanjutnya dimasukkan ke dalam kategori yang tepat.Akan lebih efesien apabila kategori berita dimasukkan secara otomatis dengan komputermenggunakan metode tertentu.

Sebelum berita dapat dikategorikan maka data berita tersebut harus diproses terlebih dahulu.Dimana dibandingkan dengan jenis data yang lain, sifat data berbentuk teks tidak terstruktur dan sulit untuk menangani. *Text mining* adalah cara agar teks dapat diolah dengan menggunakan komputer untuk menghasilkan analisis yang bermanfaat[1]. Praproses dalam *text mining* diantaranya adalah *tokenizing*, *case folding*, *stopwords*, dan *stemming*. Diantara keempat langkah tersebut yang paling penting adalah proses *stemming*yang merupakan proses menghilangkan imbuhan pada suatu kata untuk mendapatkan kata dasar dari kata tersebut.*Confix-stripping stemmer*merupakan penyempurnaan oleh Jelita Asian yang berawal dari nazief stemmer yang dibuat oleh Nazief dan Adriani.

Salah satu metode statistika yang dapat melakukan pengkategorian adalah klasifikasi. Terdapat banyak metode klasifikasi dan dalam penelitian ini akan menggunakan metode NBC dan SVM. Metode NBC telah banyak digunakan dalam penelitian mengenai text mining, beberapa kelebihan NBC diantaranya adalah algoritma sederhana tapi memiliki akurasi yang tinggi [2]. SVM teknik ini berakar pada teori pembelajaran statistik dan telah menunjukkan hasil empiris yang menjanjikan dalam berbagai aplikasi praktis dari pengenalan digit tulisan tangan sampai kategorisasi teks. SVM juga bekerja sangat baik pada data dengan banyak dimensi dan menghindari kesulitan dari permasalahan dimensionalitas [3].

Penelitian berkaitan dengan metode NBC telah dilakukan diantaranya oleh Arifiyanti (2014), menggunakan NBC dengan *confix-stripping stemmer* mendapatkan hasil ketepatan klasifikasi sebesar 86,74%. Penelitian dengan menggunakan SVM telah dilakukan oleh Liliana, Hardianto, & Ridok, (2011) menghasilkan ketepatan klasifikasi sebesar 85%. Penelitian berkaitan dengan membandingkan kedua metode NBC dan SVM telah dilakukan pada *sentiment analysis* oleh Saraswati (2011) dan Aliandu (2013), kedua penelitian mendapati hasil metode SVM lebih baik dibandingkan metode NBC. Dalam penelitian ini akan dicoba menggunakan dua metode, metode pertama adalah yang umumnya dipakai yaitu metode NBC dan metode kedua adalah metode SVM. Kedua metode tersebut akan dibandingkan, mana metode yang menghasilkan tingkat klasifikasi paling besar.

II.LANDASAN TEORI

A. Praproses Teks

Tahapan praproses ini dilakukan agar dalam klasifikasi dapat diproses dengan baik.Tahapan dalam praproses teks adalah sebagai berikut.*Case Folding*, merupakan proses untuk mengubah semua karakter pada teks menjadi huruf kecil. *Tokenizing*, merupakan proses memecah yang semula berupa kalimat menjadi kata-kata.*Stopwords*, merupakan kosakata yang bukan merupakan kata unik atau ciri pada suatu dokumen. Terakhir *Stemming*, yakni proses untuk mendapatkan kata dasar dengan cara menghilangkan awalan, akhiran, sisipan, dan *confixes* (kombinasi dari awalan dan akhiran).

B. Nazief Stemmer

Algoritma *stemming* Nazief dan Adriani dikembangkan berdasarkan aturan bahasa Indonesia yang kata-katanya menggunakan imbuhan, awalan (prefix), sisipan (infix), akhiran (suffix), dan kombinasi awalan serta akhiran

(*confixes*). Pengelompokan imbuhan *nazief stemmer* dibagi dalam beberapa kategori sebagai berikut:

a. *Inflection Suffixes*, kelompok akhiran yang tidak mengubah bentuk dari kata dasar. Kelompok ini dibagi menjadi dua, yaitu:
   - *Particle* (Partikel), termasuk di dalamnya adalah '-lah', '-kah', '-tah', dan '-pun'.
   - *Passive Pronoun* (Kata ganti kepemilikan), termasuk di dalamnya adalah '-ku', '-mu', dan '-nya'.

b. *Derivation Suffixes* (Akhiran), kumpulan akhiran yang secara langsung ditambahkan pada kata dasar. Termasuk di dalamnya adalah '-i', '-kan', dan '-an'.

c. *Derivation Prefixes* (Awalan), kumpulan awalan yang dapat ditambahkan langsung pada kata dasar yang sudah mendapatkan penambahan sampai dua awalan. Kelompok ini dibagi menjadi dua, yaitu:
   - Standar, termasuk di dalamnya adalah 'di-', 'ke-', dan 'se-'.
   - Kompleks, termasuk di dalamnya adalah 'me-', 'be-', 'pe-', dan 'te-'.

Pengelelompokan dari beberapa kategori tersebut dimodelkan sebagai berikut:

$$\left[\left[\left[\text{AW+}\right]\text{AW+}\right]\text{AW+}\right]\text{Kata Dasar}\left[\left[\text{+AK}\right]\left[\text{+KK}\right]\left[\text{+P}\right]\right]$$

Dengan:
AW = Awalan KK = Kata ganti kepunyaan
AK = Akhiran P = Partikel

Pada Tabel 1, Tabel 2, dan Tabel 3 simbol C merupakan huruf konsonan, simbol V merupakan vokal, simbol A merupakan vokal atau konsonan, dan simbol P merupakan partikel dari suatu kata, contohnya 'er'.

Tabel 1.
Aturan Pemenggalan

No	Format Kata	Pemenggalan
1	berV..	Ber-V.. | be-rV..
2	berCAP..	Ber-CAP.. dimana C!- 'r' & P!-'er'
3	berCAerV..	Ber-CaerV.. dimana C!='r'
4	Belajar	Bel-ajar
5	beC_1erC_2..	beC_1erC_2.. dimana C_1!={'r'|'l'}
6	terV..	Ter-V.. | te-rV..
7	terCerV..	Ter-CerV.. dimana C!='r'
8	terCP…	Ter-CP.. dimana C!='r' dan P!='er'
9	teC_1erC_2 ..	Te-C_1erC_2… dimana C_1!='r'
10	Me{l|r|w|y}V..	Me-{l|r|w|y}V…
11	Mem{b|f|v}…	Mem-{b|f|v}…
12	Mempe{r|l}…	Mem-pe..
13	Mem{rV|V}…	Me-m{rV|V}… | me-p {rV|V}…
14	Men{c|d|j|z}…	Men-{c|d|j|z}…
15	menV…	Me-nV.. |me-tV..
16	Meng{g|h|q}…	Meng-{g|h|q}…
17	mengV…	Meng-V… |meng-kV…
18	menyV…	Meny-sV…
19	mempV…	mempV… dimana V!='e'
20	Pe{w|y}V…	Pe-{w|y}V…
21	perV…	Per-V…|pe-rV…
22	perCAP..	Per-CAP.. dimana C!='r' dan P!='er'
23	perCAerV…	Per-CAerV…dimana C!='r'
24	Pem{b|f|V}..	Pem-{b|f|V}..
25	Pem{rV|V}…	Pe-m{rV|V}… | pe-p{rV|V}…
26	Pen{c|d|j|z}…	Pen-{c|d|j|z}…
27	penV…	Pe-nV…|pe-lV..
28	Peng{g|h|q}…	Peng-{g|h|q}…
29	pengV…	Peng-V…| peng-kV…
30	penyV…	Peny-sV…
31	pelV…	Pe-lV… kecuali 'pelajar'menjadi 'ajar'
32	peCerV…	Per-erV…dimana C!={r|w|y|l|m|n}
33	peCP…	peCP… dimana C!={r|w|y|l|m|n} dan P!='er'

C. Confix-Stripping Stemmer

Pada tahun 2007 algoritma *nazief stemmer* kemudiandikembangkan lagi oleh Jelita Asian, dengan menambahkan beberapa perbaikan yang bertujuan untuk meningkatkan hasil stemming yang diperoleh.Algoritma ini kemudian dikenal sebagai *confix-stripping stemmer*. Perbaikan tersebut antara lain sebagai berikut:

1. Menggunakan kamus kata dasar yang lebih lengkap.
2. Memodifikasi dan menambahkan aturan pemenggalan untuk tipe awalan yang kompleks (memodifikasi aturan pada Tabel 1 sesuai modifikasi pada Tabel 2 dan menambahkan aturan pada Tabel 3 ke dalam Tabel 1)
3. Menambahkan aturan *stemming* untuk kata ulang dan bentuk jamak, misalnya kata 'buku-buku' yang menjadi 'buku'. Hal ini dilakukan dengan melakukan pemisahan kata tersebut menjadi dua kata yang masing-masing di-*stemming*. Jika *stemming* memberikan kata dasar yang sama, maka keluaran kata dasarnya adalah hasil *stemming* tersebut. Jika hasil *stemming* dua kata tersebut berbeda maka disimpulkan bahwa masukan adalah kata ulang semu dan tidak memiliki bentuk kata dasar lagi.
4. Aturan*rule precedence*penghilangan awalan dilakukan terlebih dahulu kemudian diikuti oleh penghilangan akhiran dan berlaku jika kata memiliki kombinasi awalan-akhiran 'be-lah', 'be-an', 'me-i', 'di-i', 'pe-i', atau 'te-i', misalnya 'bertaburan', 'melindungi', 'dilengkapi', dan 'teradili'.

Tabel 2.
Modifikasi Aturan

Aturan	Format Kata	Pemenggalan
12	Mempe…	Mem-pe…
16	Meng{g|h|q|k}	Meng-{g|h|q|k}

Tabel 3.
Aturan Tambahan

Aturan	Format Kata	Pemenggalan
34	terC_1erC_2…	terC_1erC_2… dimana C_1!='r'
35	peC_1erC_2…	peC_1erC_2…dimana C_1!={r|w|y|l|m|n}

D. Naive Bayes Classifier

Teorema Bayes merupakan teorema yang mengacu konsep probabilitas bersyarat[3].Secara umum teorema Bayes dapat dinotasikan pada persamaan berikut:

$$P(A|B)=\frac{P(A)P(B|A)}{P(B)} \quad (1)$$

Metode *naivebayesclassification* (NBC), merupakan salah satu metode yang dapat mengklasifikasikan teks.Kelebihan NBC adalah algoritmanya sederhana tetapi memiliki akurasi yang tinggi. Dalam algoritma NBC setiap dokumen direpresentasikan dengan pasangan atribut "a_1, a_2, a_3,..., a_n" dimana a_1 adalah kata pertama, a_2 adalah kata kedua dan seterusnya. Sedangkan V adalah himpunan kategori berita. Pada saat klasifikasi algoritma akan mencari probabilitas

tertinggi dari semua kategori dokumen yang diujikan (V_{MAP}). Adapun persamaan V_{MAP} adalah sebagai berikut:

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}}\ P(v_j)\prod_i P(a_i \mid v_j) \quad (2)$$

Nilai $P(v_j)$ dihitung pada saat data *training*, didapat dengan rumus sebagai berikut:

$$P(v_j) = \frac{|doc\ j|}{|training|} \quad (3)$$

Dimana $|doc\ j|$merupakan jumlah dokumen (artikel berita) yang memiliki kategori *j* dalam *training*.Sedangkan $|training|$ merupakan jumlah dokumen (artikel berita) dalam contoh yang digunakan untuk *training*. Untuk probabilitas kata a_i untuk setiap kategori $P(a_i|v_j)$, dihitung pada saat *training*.

$$P(a_i \mid v_j) = \frac{|n_i + 1|}{|n + kosakata|} \quad (4)$$

Dimana n_i adalah jumlah kemunculan kata a_i dalam dokumen yang berkategori v_j, sedangkan *n* adalah banyaknya seluruh kata dalam dokumen dengan kategori $v_j$dan $|kosakata|$ adalah banyaknya kata dalam contoh pelatihan.

E. Term Frequency Inverse Document Frequency

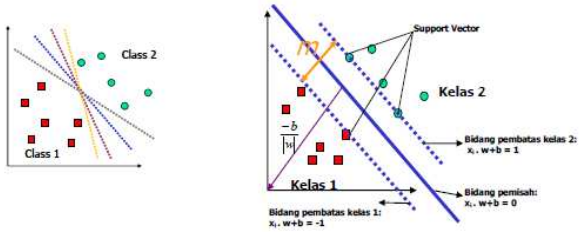
Term Frequency Inverse Document Frequency (TF-IDF) merupakan pembobot yang dilakukan setelah ekstrasi artikel berita. Rumus dalam menemukan pembobot dengan TF-IDF adalah sebagai berikut :

$$w_{ij} = tf_{ij} \times idf\ ,\ idf = \log\left(\frac{N}{df_j}\right) \quad (5)$$

Dimana w_{ij} adalah bobot dari kata *i* pada artikel ke *j*, *N* merupakan jumlah seluruh dokumen, tf_{ij} adalah jumlah kemunculan kata i pada dokumen *j*, df_j adalah jumlah artikel *j* yang mengandung kata *i*. TF-IDF dilakukan agar data dapat dianalisis dengan menggunakan *support vector machine*.

F. Support Vector Machine

SupportVectorMachine(SVM) adalahsistem pembelajaranyang menggunakanhipotesisfungsilineardalam ruang berdimensi tinggidan dilatihdenganalgoritmaberdasarkanteori optimasidengan menerapkan*learning bias* yangberasal dariteori statistik[4].Tujuan utama darimetode iniadalah untuk membangunOSH(*Optimal Separating Hyperplane*), yang membuatfungsipemisahanoptimumyang dapat digunakanuntuk klasifikasi.



Gambar 1 Alternatif bidang pemisah (kiri) dan bidang pemisah terbaik dengan margin (m) terbesar (kanan).

Data yang berada pada bidang pembatas disebut dengan *support vector*. Dalam Gambar 1, dua kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. $|b|/\|\boldsymbol{w}\|$merupakan jarak bidang pemisah yang tegak lurus dari titik pusat koordinat dan$\|\boldsymbol{w}\|$ adalah jarak *euclidean* dari **w**. Bidang pembatas pertama membatasi kelas pertama sedangkan bidang pembatas kedua membatasi kelas kedua, sehingga diperoleh:

$$\begin{aligned} \mathbf{x_i w} + b &\geq +1,\ y_i = +1 \\ \mathbf{x_i w} + b &\leq -1,\ y_i = -1 \end{aligned} \quad (6)$$

w adalah normal bidang dan *b* adalah posisi bidang alternatif terhadap pusat koordinat. Nilai margin (jarak) antara bidang pembatas (berdasarkan rumus jarak garis ke titik pusat) adalah $\frac{1-b-(-1-b)}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$. Nilai margin ini dimaksimalkan dengan tetap memenuhi persamaan (6). Dengan mengalikan *b* dan **w** dengan sebuah konstanta, akan dihasilkan nilai margin yang dikalikan dengan konstata yang sama. Oleh karena itu, *constraint* pada persamaan (6) merupakan *scaling constraint* yang dapat dipenuhi dengan *rescaling b* dan **w**. Selain itu karena memaksimalkan $1/\|\boldsymbol{w}\|$ sama dengan meminimumkan$\|\boldsymbol{w}\|^2$. Jika kedua bidang pembatas pada persamaan (6) direpresentasikan dalam pertidaksamaan,

$$y_i\left(\mathbf{x_i w} + b\right) - 1 \geq 0 \quad (7)$$

maka pencarian bidang pemisah terbaik dengan nilai margin terbesar dapat dirumuskan menjadi masalah optimasi konstrain, yaitu:

$$\begin{aligned} &\min \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{dengan } y_i\left(\mathbf{x_i w} + b\right) - 1 \geq 0 \end{aligned} \quad (8)$$

Untuk mengklasifikasikan data yang tidak dapat dipisahkan secara linier formula SVM ditambahkan variabel ξ_i sering disebut dengan *soft margin hyperplane*. Dengan demikian formula pencarian bidang pemisah terbaik berubah menjadi:

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + C\left(\sum_{i=1}^{n}\xi_i\right) \quad (9)$$

$$\text{Dengan } y_i(\mathbf{x_i} + b) \geq 1 - \xi_i\ ,\ \xi_i \geq 0$$

C adalah parameter yang menentukan besar penalti akibat kesalahan dalam klasifikasi data dan nilainya ditentukan oleh pengguna.Sehingga peran dari *C* adalah meminimalkan kesalahan pelatihan dan mengurangi kompleksitas model.Untuk kasus data dengan kategori lebih dari 2 atau *multiclass*, digunakan metode *One Against One* (OAO).

G. Pengukuran Performa

Pengukuran performa dilakukan untuk melihat hasil yang didapatkan dari klasifikasi. Terdapat beberapa cara untuk mengukur performa, beberapa cara yang sering digunakan adalah dengan menghitung akurasi, *recall, precission* dan*F-measure*[5].

$$akurasi = \frac{jumlah\ klasifikasi\ benar}{jumlah\ dokumen\ uji\ coba} \times 100\%$$

$$recall = \frac{\left|\{relevant\ doc\} \cap \{retrieved\ doc\}\right|}{\left\{\left|relevant\ doc\right|\right\}}$$

$$precision = \frac{\left|\{relevant\ doc\} \cap \{retrieved\ doc\}\right|}{\left\{\left|retrieved\ doc\right|\right\}}$$

$$F = \frac{2 \times recall \times precision}{recall + precision}$$

III. METODOLOGI PENELITIAN

A. Sumber Data

Sumber data yang akan digunakan dalam penelitian ini adalah artikel berita pada koran online kompas.com yang

terdiri dari 12 artikel. Kategori tersebut adalah berita nasional, internasional, olahraga, sains, edukasi, ekonomi, tekno, entertaintment, otomotif, health, properti, dan travel. Tiap kategori akan diambil sebanyak 100 artikel sehingga data artikel keseluruhan berjumlah 1200.

B. Langkah Analisis

1. Menyiapkan data artikel, daftar *stopwords*, dan kata dasar.
   - Artikel berita*online*Januari hingga Desember tahun 2014. Data sampel tersebut dibagi menjadi data *training* dan data *testing* dengan proporsi 70:30.
   - Daftar *stopwords*, didapatkan pada tesis F. Tala yang berjudul "*A Study of Stemming Effect on Information Retrieval in Bahasa Indonesia*".
   - Kata dasar dari kamus besar bahasa Indonesia.
2. Praproses Teks
   a) Melakukan *case folding,*proses untuk mengubah semua karakter pada teks menjadi huruf kecil.
   b) *Tokenizing* untuk memecah kalimat menjadi kata per kata.
   c) Melakukan *stemming* pada kata-kata yang tersisa pada dokumen teks untuk mendapatkan kata dasar. Pada tahap ini dilakukan algoritma *confix-stripping stemmer* untuk mendapatkan kata dasar.
   d) Kemudian dilakukan proses *stopping* berdasarkan *stoplist* yang berisi *stopwords* yang telah ditentukan sebelumnya.
3. Klasifikasi teks menggunakan NBC dengan tahapan
   a) Membagi data menjadi *testing* dan *training*, pada data *training* telah diketahui jenis dari kategori berita.
   b) Menghitung probabilitas dari V_j, dimana V_j merupakan kagetori berita, yaitu j_1 = nasional, j_2 = internasional, dan seterusnya.
   c) Menghitung probabilitas kata w_k pada kategori v_j.
   d) Model probabilitas NBC disimpan dan digunakan untuk tahap data *testing*.
   e) Menghitung probabilitas tertinggi dari semua kategori yang diujikan (V_{MAP}).
   f) Mencari nilai V_{MAP} paling maksimum dan memasukkan artikel berita tersebut pada kategori dengan V_{MAP} maksimum.
   g) Menghitung nilai akurasi dari model yang terbentuk.
4. Klasifikasi teks menggunakan SVM dengan tahapan
   a) Membagi data menjadi *testing* dan *training*, pada data *training* telah diketahui jenis dari kategori berita.
   b) Merubah teks menjadi vektor dan pembobotan kata dengan *tf-idf*.
   c) Menentukan pembobot parameter pada SVM tiap jenis kernel.
   d) Membangun model SVM menggunakan fungsi *Radial Basis Function* dan linier.
   e) Menghitung nilai akurasi dari model yang terbentuk.
5. Membandingkan performansi metode NBC dan SVM berdasarkan tingkat akurasi ketepatan klasifikasi.

IV. HASIL DAN PEMBAHASAN

Dalam pembahasan ini data telah dibagi menjadi dua yaitu data *training* dan *testing* dengan proporsi 70:30. Jumlah *word vector* yang akan diuji coba pada data *training* adalah 1000,1500, 2000, 2500, 3000, 3500, 4000, 4500, dan 10000. Sedangkan untuk data *testing* karena jumlah artikel berita lebih sedikit maka *word vector* yang akan digunakan adalah 1000,1500, 2000, 2500, dan 3000.

A. Naïve Bayes Classifier

Pada data *training* kelas pada artikel berita telah diketahui sebelumnya. Dimana tujuan data *training* adalah untuk menghasilkan model dari *Naïve Bayes Classifier* (NBC) untuk mengetahui ketepatan klasifikasi, selain itu pada data *training* juga memperhatikan waktu yang diperlukan pada pembentukan model.Berikut merupakan hasil dari data *training*.

Tabel 4.
Ketepatan Klasifikasi (%) Pembentukan Model NBC Menggunakan Data *Training*

Word Vector	Ketepatan Klasifikasi	Word Vector	Ketepatan Klasifikasi
500	90,2381	3500	94,7619
1000	92,619	4000	94,881
1500	93,6905	4500	95,119
2000	94,4048	6000	95,2381
2500	94,7619	**10000**	**95,883**
3000	94,6429		

Tabel 4 yang bercetak tebal memperlihatkan bahwa dengan menggunakan *word vector* sebanyak 10000 akan menghasilkan tingkat klasifikasi yang paling baik. Ketepatan klasifikasi cenderung meningkat.Kecuali pada *word vector* 3000 dimana ketepatan sempat turun, kemudian terus meningkat hingga pada *word vector* terakhir.

Selanjutnya adalah menguji masing-masing model pada *word vector* tersebut dengan menggunakan data *testing*.Berikut merupakan hasil klasifikasi berita dengan data *testing* menggunakan model yang telah terbentuk sebelumnya.

Tabel 5.
Ketepatan Klasifikasi (%) NBC Menggunakan Data *Testing*

Word Vector		Ketepatan Klasifikasi	Word Vector		Ketepatan Klasifikasi
Data Training	**Data Testing**		**Data Training**	**Data Testing**	
500	500	72,5	3500	3000	81,67
1000	1000	75,55	**4000**	**3000**	**82,22**
1500	1500	76,94	**4500**	**3000**	**82,22**
2000	2000	79,17	**6000**	**3000**	**82,22**
2500	2500	80,28	**10000**	**3000**	**82,22**
3000	3000	81,39			

Berdasarkan Tabel 5 akurasi yang dicetak tebal merupakan akurasi tertinggi untuk ketepatan prediksi artikel berita.memperlihatkan bahwa ketepatan klasifikasi untuk *word vector* 4000, 4500, 6000 dan 10000 memberikan hasil yang terbaik dan menghasilkan hasil yang sama yaitu sebesar 82,2222%.

Tabel 6.
Hasil Akurasi, *Precision, Recall*, dan *F-Measure* NBC pada Data *Testing*

Kategori	**Akurasi**	**Precision**	**Recall**	**F-Measure**
Nasional	73,30%	75,90%	73,30%	74,60%
Internasional	80,00%	72,70%	80,00%	76,20%
Olahraga	80,00%	96,00%	80,00%	87,30%
Sains	80,00%	75,00%	80,00%	77,40%
Edukasi	93,30%	77,80%	93,30%	84,80%
Ekonomi	76,70%	76,70%	76,70%	76,70%
Tekno	76,70%	100,00%	76,70%	86,80%
Entertaiment	90,00%	96,40%	90,00%	93,10%
Otomotif	83,30%	100,00%	83,30%	90,90%
Health	93,30%	87,50%	93,30%	90,30%
Properti	66,70%	83,30%	66,70%	74,10%

Travel	93,30%	65,10%	93,30%	76,70%
Rata-rata	**82,20%**	**83,90%**	**82,20%**	**82,40%**

Berdasarkan rata-rata akurasi, *recall, precision,* dan *F-Measure*pada Tabel 6 memperlihatkan hasil yang cukup baik. Masing-masing nilainya adalah 82,2%, 83,9%, 82,2%, dan 82,4%. Untuk tingkat akurasi paling tinggi dihasilkan oleh kategori berita edukasi, health, dan travel dengan nilai akurasi 93,3%. Berbeda dengan tiga kategori tersebut kategori berita properti menghasilkan akurasi yang paling rendah yaitu 66,7%. Untuk ukuran *precision* kategori tekno dan otomotif bernilai 100% sebaliknya travel menjadi yang paling rendah yaitu 65,1%. *Recall* paling tinggi terdapat pada kategori edukasi, health, dan travel sedangkan untuk paling rendah adalah properti. Untuk ukuran gabungan dari *precision* dan *recall* yaitu *F-Measure* memperlihatkan bahwa kategori entertaiment adalah yang paling tinggi sedangkan yang paling rendah adalah properti.

Untuk kategori berita tekno dan otomotif tidak terdapat kategori berita lain yang diprediksikan masuk ke dalam kedua kategori tersebut. Tidak adanya kategori lain yang masuk, menyebabkan kategori tekno dan otomotif memiliki nilai *precision* sebesar 100%. Dalam pembahasan menggunakan NBC berarti terdapat kata yang berbeda pada saat menggunakan data *training*. Sehingga saat artikel berita pada data *testing* dicoba, hanya artikel berita yang memiliki kata berbeda tersebut yang akan diprediksi ke dalam kedua kategori tersebut. Sebaliknya seperti pada Tabel 6 travel menjadi kategori dengan nilai *precision* rendah. Kata yang terdapat pada data *training* artikel berita travel juga dipakai pada artikel berita lainnya sehingga berita kategori lain dapat diprediksi masuk kategori travel.Kesalahan klasifikasi yang terjadi dapat dikurangi dengan menambahkan data *training* yang memiliki kata yang lebih representatif, sehingga kata pada masing-masing kategori dapat lebih akurat untuk diprediksi.

B. Support Vector Machine

Sama seperti pembahasan pada NBC mulanya data *training* dibagi menjadi 10 *word vector*. Tiap *word vector* dicari ketepatan klasifikasi yang paling baik dengan menggunakan parameter *Support Vector Machine* (SVM) yaitu *C*dengan batasan *C* akan dari 10^{-2} hingga 10^{4}[6]. Sedangkan *gamma*akanditentukan melalui percobaan untuk mendapatkan hasil *training* yang paling baik. Untuk mendapatkan model pada SVM kernel RBFakan digunakan seluruh data *training*. Berikut merupakan hasil percobaan pada data *training*dengan *word vector* 10000.

Tabel 7.
Ketepatan Klasifikasi SVM Kernel RBF Menggunakan Data *Training Word Vector* 10000

gamma		1000	100	10	1	0.1	0.01	0.001
C	0.01	100	100	100	100	100	100	91.79
	0.1	100	100	100	100	100	100	91.79
	1	100	100	100	100	100	100	99.76
	10	100	100	100	100	100	100	100
	100	100	100	100	100	100	100	100
	1000	100	100	100	100	100	100	100
	10000	100	100	100	100	100	100	100
gamma		0.0001	7.96E-05	1E-05	1E-06	1E-07	1E-08	1E-09
C	0.01	88.93	89.64	92.5	92.74	92.74	92.86	93.33
	0.1	88.93	89.64	92.5	92.74	92.74	92.86	93.33
	1	88.93	89.64	92.5	92.74	92.74	92.86	93.33
	10	99.88	99.64	92.5	92.74	92.74	92.86	93.33
	100	100	100	99.88	92.74	92.74	92.86	93.33
	1000	100	100	**100**	92.74	92.74	92.86	93.33
	10000	100	100	100	100	100	100	93.33

Berdasarkan Tabel 7dapat dilihat bahwa dengan menggunakan parameter *C* dan *gamma* pada percobaan SVM akan mempengaruhi hasil ketepatan klasifikasi berita pada data training. *Gamma* 1000 hingga 0,1 didapatkan ketepatan klasifikasi pada data training sebesar 100% pada tiap parameter *C*. Nilai *gamma* yang semakin mengecil mulai mempengaruhi ketepatan klasifikasi seperti yang terlihat pada gamma 0,001 dimana untuk *C* 10^{-2}hingga 10^{0} ketepatan klasifikasi dibawah 100%. Semakin mengecil nilai gamma semakin mengurangi ketepatan klasifikasi sehingga perlu ditambahkan parameter *C* yang lebih besar.Menggunakan cara yang sama maka didapatkan parameter untuk tiap *word vector*.

Tabel 8.
Parameter RBF yang Digunakan Pada Data Training

word vector	**1000**	**1500**	**2000**	**2500**	**3000**
gamma	0.000998	0.0006579	0.0001	0.00001	0.0003237
C	100	100	100	1000	100
word vector	**3500**	**4000**	**4500**	**6000**	**10000**
gamma	0.00028	0.0001	0.00001	0.00001	0.00001
C	100	100	1000	1000	1000

Paramater yang ada pada Berdasarkan Tabel 7dapat dilihat bahwa dengan menggunakan parameter *C* dan *gamma* pada percobaan SVM akan mempengaruhi hasil ketepatan klasifikasi berita pada data training. *Gamma* 1000 hingga 0,1 didapatkan ketepatan klasifikasi pada data training sebesar 100% pada tiap parameter *C*. Nilai *gamma* yang semakin mengecil mulai mempengaruhi ketepatan klasifikasi seperti yang terlihat pada gamma 0,001 dimana untuk *C* 10^{-2}hingga 10^{0} ketepatan klasifikasi dibawah 100%. Semakin mengecil nilai gamma semakin mengurangi ketepatan klasifikasi sehingga perlu ditambahkan parameter *C* yang lebih besar.Menggunakan cara yang sama maka didapatkan parameter untuk tiap *word vector*.

Tabel 8merupakan parameter yang menghasilkan ketepatan klasifikasi 100% pada data training dan menghasilkan ketepatan klasifikasi yang paling tinggi pada data testing yang akan dibahas selanjutnya.

Tabel 9.
Ketepatan dan Waktu Klasifikasi SVM Kernel Linier Menggunakan Data *Training*

C		**Ketepatan Klasifikasi (%)**						
		0.01	0.1	**1**	10	100	1000	10000
Word Vector	1000	99.881	100	**100**	100	100	100	100
	1500	99.881	100	**100**	100	100	100	100
	2000	99.881	100	**100**	100	100	100	100
	2500	100	100	**100**	100	100	100	100
	3000	100	100	**100**	100	100	100	100
	3500	100	100	**100**	100	100	100	100
	4000	100	100	**100**	100	100	100	100
	4500	100	100	**100**	100	100	100	100
	6000	100	100	**100**	100	100	100	100
	10000	100	100	**100**	100	100	100	100

Berdasarkan Tabel 9memperlihatkan untuk SVM dengan menggunkan kernel linier untuk setiap *word vector* pada data *training* didapatkan nilai ketepatan sebesar 100%. Kecuali untuk *word vector* 1000, 1500, dan 2000 pada *c*=0,01.Setelah mengetahui bahwa ketepatan klasifikasi pada data *training* baik untuk SVM dengan kernel RBF maupun

kernel linier, maka selanjutnya akan masuk tahap dengan menggunakan data *testing* untuk tiap kernel.

Tabel 10.
Ketepatan Klasifikasi SVM Menggunakan Data *Testing*

Word Vector		KetepatanKlasifikasi (%)	
Training	Testing	RBF	Linier
1000	1000	**79,4444**	78,6111
1500	1500	**83,8888**	83,6111
2000	2000	**86,3889**	85,5556
2500	2500	86,3889	86,3889
3000	3000	86,1111	86,1111
3500	3000	85,8333	85,8333
4000	3000	**86,3889**	86,1111
4500	3000	86,6667	86,6667
6000	3000	87,5	87,5
10000	3000	88,0556	88,0556

Tabel 10menunjukkan untuk kernel RBF bahwa pada saat *word vector* 1000, 1500, 2000, dan 4000 menghasilkan ketepatan klasifikasi yang lebih tinggi dibandingkan dengan kernel linier. Saat jumlah *word vector* ditambahkan menjadi 4500 hingga 10000 ketepatan klasifikasi antara kernel RBF maupun linier menghasilkan ketepatan klasifikasi yang sama. Pada umumnya RBF akan lebih baik jika variabel lebih dari 1000[7], namun dalam penelitian yang dilakukan didapatkan bahwa pada *word vector* 4500 ke atas linier sama baiknya dengan kernel RBF. Kernel linier akan sama baiknya dengan kernel yang lebih rumit jika memiliki *input space* yang cukup. Ini berarti kategori teks terpisah secara linier di ruang fitur.

Kemudian untuk pengukuran performa pada SVM akan digunakan linier SVM pada *word vector* 10000 untuk selanjutnya dibandingkan dengan hasil pada NBC.

Tabel 11.
Hasil Akurasi, *Precision, Recall*, dan *F-Measure* SVM pada Data *Testing*

Kategori	Akurasi	Precision	Recall	F-Measure
Nasional	86.7%	81.3%	86.7%	83.9%
Internasional	90.0%	73.0%	90.0%	80.6%
Olahraga	86.7%	89.7%	86.7%	88.1%
Sains	80.0%	82.8%	80.0%	81.4%
Edukasi	86.7%	96.3%	86.7%	91.2%
Ekonomi	90.0%	73.0%	90.0%	80.6%
Tekno	83.3%	100.0%	83.3%	90.9%
Entertaiment	96.7%	90.6%	96.7%	93.5%
Otomotif	86.7%	100.0%	86.7%	92.9%
Health	93.3%	96.6%	93.3%	94.9%
Properti	93.3%	93.3%	93.3%	93.3%
Travel	83.3%	92.6%	83.3%	87.7%
Rata-rata	88.1%	89.1%	88.1%	88.3%

Hasil klasifikasi data *testing* menggunakan SVM linier pada Tabel 11 menunjukkan performa yang cukup baik dengan masing-masing nilai dari akurasi, *precision*, *recall*, dan *F-Measure* adalah 88,1%, 89,1%, 88,1%, dan 88,3%. Kategori berita entertaiment menjadi kategori dengan tingkat akurasi yang paling tinggi yaitu 96,7%, sebaliknya sains menjadi kategori dengan tingkat akurasi yang paling rendah yaitu 80,0%. Untuk precision dengan nilai paling baik adalah kategori tekno dan otomotif sebesar 100%, sedangkan kategori internasional dan ekonomi adalah kategori dengan precision terendah sebesar 73,0%. Hasil recall tertinggi adalah kategori entertaiment dengan nilai sebesar 96,7% dan terendah adalah kategori sains dengan nilai sebesar 80,0%. Nilai F-Measure menunjukkan performa yang paling baik adalah kategori health 94,9%, sedangkan yang paling rendah adalah kategori internasional dan ekonomi 80,6%.

C. Perbandingan Antara NBC dan SVM

Tabel 12.
Perbandingan Ketepatan Klasifikasi Antara NBC dan SVM

Metode	Akurasi	Precision	Recall	F-Measure
NBC	82,2%	83,9%	82,2%	82,4%
SVM	88.1%	89.1%	88.1%	88.3%

D. Melihat hasil dari Perbandingan Antara NBC dan SVM

Tabel 12maka untuk semua cara pengukuran performa baik akurasi, *precision, recall,* dan *F-Measure* SVM kernel linier lebih baik dari NBC. Selain itu secara waktu saat menggunkan aplikasi SVM jauh lebih cepat untuk mendapatkan hasil daripada NBC.Secara keseluruhan terdapat 33 berita yang tidak bisa diprediksi dengan baik oleh kedua metode.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Setelah sebelumnya didapatkan hasil dan pembahasan untuk klasifikasi berita Indonesia menggunakan metode NBC dan SVM dengan *confix stripping stemmer*.Berikut merupakan kesimpulan yang didapatkan.

1. Metode *Naive Bayes Classifier* dapat melakukan klasifikasi berita Indonesia cukup baik. Hasil yang didapatkan pada saat data *testing* pada masing-masing pengukuran performa akurasi, *precision, recall,* dan *F-Measure* sebesar 82,2%; 83,9%; 82,2%; dan 82,4%.
2. Metode *Support Vector Machine* antara kernel RBF dan kernel linier pada *word vector* 10000sama baiknya dalam melakukan klasifikasi berita Indonesia. Menggunakan data *testing* didapatkan untuk tiap pengukuran performa akurasi, *precision, recall,* dan *F-Measure* adalah 88,1%, 89,1%, 88,1%, dan 88,3%.
3. Perbandingan antara kedua metode NBC dan SVM didapatkan hasil SVM kernel RBF dan linier lebih baik dibandingkan dengan NBC.

B. Saran

Saran untuk penelitian yang akan datang adalah.

1. Dalam penelitian klasifikasi berita ini tidak melakukan pemilihan atribut/variabel. Sehingga untuk penelitian selanjutnya dapat dilakukan pemilihan atribut untuk mengurangi jumlah data.
2. Dalam prediksi kelas pada *multiclass* SVM hanya menggunakan metode *one against one* dimana terdapat metode lainnya seperti *one against all*.

DAFTAR PUSTAKA

[1] Ian H Witten, Eibe Frank, and Mark A Hall, *Data Mining Practical Machine Learning Tools and Techniques*. USA: Elsevier, 2011.

[2] I Rish, "An empirical study of The Naive Bayes Classifier," *International Joint Conference on Artificial Intelligence*, 2006.

[3] P N Tan, M Steinbach, and V Kumar, *Introduction to Data Mining*. Boston: Pearson Education, 2006.

[4] N Christianini and J Shawe-Taylor, *An Introduction to Support Vector Machines*. UK, Cambridge: Cambridge University Press, 2000.

[5] Andreas Hotho, Andreas Nurnberger, and Gerhard Paass, *A Brief Survey of Text Mining*. Kassel: University of Kassel, 2005.

[6] Chien-Ming Huang, Yuh-Jye Lee, Dennis K.J Lin, and Su-Yun Huang, "Model Selection For Support Vector Machines Via Uniform Design," *Computational Statistics & Data Analysis*, pp. 335-346, 2007.

[7] Neelima Guduru, *Text Mining With Support Vector Machines And Non-Negative Matrix Factorization Algorithms*.: University Of Rhode Island, 2006.